

Interoperability 101: OGC Standards

Karl Benedict
Earth Data Analysis Center
Department of Geography
University of New Mexico
kbene@edac.unm.edu

Overview

- **Motivating Scenarios**
- **Relevant OGC Standards**
- **Review**

What are we trying to
do?

Visualize Geospatial Data

- Potentially large (multi-TB) data collection
- Temporal Component?
- Delivery to diverse clients
 - Desktop
 - Mobile



Deliver or Access Data

- Actual data values are needed - not just visualizations of data
- Process before delivery?
- Raster (gridded) and Vector source data types
- Delivery in a variety of formats



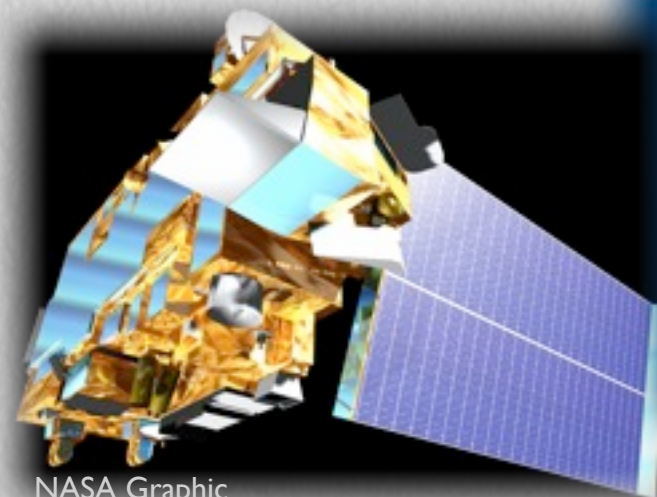
Advertisement or Discovery of Data

- Efficient discovery of specific data products among many candidates
- Participation in or use of distributed search tools
- Discovery based upon data characteristics and uses documented in metadata



Interaction with Sensors

- Documenting sensor characteristics
- Tasking sensors
- Monitoring sensor status
- Retrieving data from sensors in an efficient and “smart” manner



NASA Graphic



There's an OGC Standard for That

- Visualization: Web Map Service (WMS)
- Data Download: Web Feature and Web Coverage Service (WFS & WCS)
- Data Discovery: Catalog Services for Web (CSW)
- Sensor Interactions: Sensor Web Enablement Standards (SWE)

The Standards

Common Elements

- HTTP Get and/or Post request model



Common Elements

- HTTP Get and/or Post request model
- GetCapabilities request



Common Elements

- HTTP Get and/or Post request model
- GetCapabilities request

Google – “request=GetCapabilities”



Common Elements

- HTTP Get and/or Post request model
- GetCapabilities request
- Standard error reporting

```
<?xml version="1.0" encoding="UTF-8"?>  
<xsd:schema  
  targetNamespace="http://www.opengis.net/ogc"  
  xmlns:ogc="http://www.opengis.net/ogc"  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  elementFormDefault="qualified">  
  
  <xsd:element name="ServiceExceptionReport">  
    <xsd:complexType>
```


Common Elements

- HTTP Get and/or Post request model
- GetCapabilities request
- Standard error reporting
- OGC Web Services Common specification (OWS)

Common Elements

- HTTP Get and/or Post request model
- GetCapabilities request
- Standard error reporting
- OGC Web Services Common specification (OWS)
- Get* requests for additional products

Web Map Service WMS

- Delivers web-friendly map images
- Basic location-based query
- Explicit support for temporal requests
- Separate Earth Observation Profile
- Separate Web Map Tile Service specification
- Requests
 - GetMap - map image
 - GetFeatureInfo - attribute information for specific location

Web Feature Service

WFS

- Delivers features and associated attributes in GML and other formats
- Includes filter language for feature selection/extraction
- (optional) Feature updates
- Requests
 - DescribeFeatureType - layers metadata
 - GetFeature - retrieve feature(s)
 - GetGmlObject (optional) - retrieve feature(s) by ID
 - LockFeature (optional) - lock feature for modification
 - Transaction (optional) - perform feature creation/modification

Web Coverage Service WCS

- Delivers data, commonly gridded, in a variety of formats
- Explicit support for temporal requests
- Support for spatial, temporal, and range subsetting; spatial interpolation.
- Requests
 - DescribeCoverage - detailed layer metadata
 - GetCoverage - retrieve specified coverage

Catalog Services for Web - CSW

- Definition of a common catalog query language extensible by specific user communities
- Specification of both session-based and stateless query models, emphasis on session-based
- Protocol binding into z39.50, CORBA, HTTP
- Requests
 - Discovery: DescribeRecordType, GetDomain, “query”*, “present”*
 - Session Management: “initialize”, “close”, “status”, “cancel”
 - Catalog Management: HarvestResource, “transaction”
 - Brokered Access: “order”

* limited to stateful interactions

Sensor Web Enablement Standards - SWE

- **Observations & Measurements (O&M)** - Standard models and XML Schema for encoding observations and measurements from a sensor, both archived and real-time.
- **Sensor Model Language (SensorML)** - Standard models and XML Schema for describing sensors systems and processes associated with sensor observations; provides information needed for discovery of sensors, location of sensor observations, processing of low-level sensor observations, and listing of taskable properties, as well as supports on-demand processing of sensor observations.
- **Transducer Model Language (TransducerML or TML)** - The conceptual model and XML Schema for describing transducers and supporting real-time streaming of data to and from sensor systems.
- **Sensor Observations Service (SOS)** - Standard web service interface for requesting, filtering, and retrieving observations and sensor system information. This is the intermediary between a client and an observation repository or near real-time sensor channel.
- **Sensor Planning Service (SPS)** - Standard web service interface for requesting user-driven acquisitions and observations. This is the intermediary between a client and a sensor collection management environment.
- **Sensor Alert Service (SAS)** - Standard web service interface for publishing and subscribing to alerts from sensors.
- **Web Notification Services (WNS)** - Standard web service interface for asynchronous delivery of messages or alerts from SAS and SPS web services and other elements of service workflows.

from OGC SWE WG Entry Page:

<http://www.opengeospatial.org/projects/groups/sensorweb>

Review

- Data Visualization: WMS
- Data Access/Delivery: WFS, WCS
- Data Discovery: CSW
- Sensor Interaction: SWE family of standards

Resources

- **Standards** (<http://www.opengeospatial.org/standards>)
 - **WMS** (<http://www.opengeospatial.org/standards/wms>)
 - **WFS** (<http://www.opengeospatial.org/standards/wfs>)
 - **WCS** (<http://www.opengeospatial.org/standards/wcs>)
 - **CSW** (<http://www.opengeospatial.org/standards/specifications/catalog>)
- **SWE Overview White Paper:**
http://portal.opengeospatial.org/files/?artifact_id=39467

Questions?